

IN THE CLAIMS

Please amend the claims as follows:

Claim 1 (Original): A lossless coding method for a signal in a floating-point format, comprising:

(a) a step of converting a first signal sample sequence in the floating-point format into a second signal sample sequence in an integer format;

(b) a step of losslessly compressing said second signal sample sequence in the integer format to produce a first code sequence;

(c) a step of producing a difference signal in the floating-point format that corresponds to the difference between said second signal sample sequence in the integer format and said first signal sample sequence in the floating-point format;

(d) a step of determining which coding processing is of higher compression efficiency, a first coding processing that codes a range of bits capable of assuming non-zero in a sample of said difference signal in the floating-point format, the range of bits being determined by the bit position of the most significant “1” in the corresponding sample in said second signal sample sequence in the integer format, or a second coding processing that directly codes said difference signal; and

(e) a step of coding said difference signal by said first coding processing to produce a second code sequence if the result of said determination indicates said first coding processing, or coding said difference signal by said second coding processing to produce a second code sequence if the result of said determination indicates said second coding processing.

Claim 2 (Original): A lossless coding method for a signal in a floating-point format, comprising:

(a) a step of determining which compression is of higher compression efficiency, a direct lossless compression of a first signal sample sequence in the floating-point format on a frame basis or a dual-signal separate compression of an integer-value sample sequence and a difference signal derived from the first signal sample sequence;

(b) a step of performing the dual-signal separate compression if the result of said determination indicates the dual-signal separate compression, the step (b) comprising:

(b-1) a step of converting the first signal sample sequence into a second signal sample sequence in an integer format;

(b-2) a step of losslessly compressing said second signal sample sequence in the integer format to produce a first code sequence;

(b-3) a step of producing a difference signal in the floating-point format that corresponds to the difference between said second signal sample sequence in the integer format and said first signal sample sequence in the floating-point format;

(b-4) a step of producing a second code sequence from said difference signal in the floating-point format;

(c) a step of directly losslessly compressing said first signal sample sequence to produce a third code sequence if the result of said determination indicates the direct compression; and

(d) a step of producing an auxiliary code that indicates whether said direct lossless compression or said separate compression of two types of signals is performed.

Claim 3 (Original): A lossless coding method for a signal in a floating-point format according to Claim 2, in which, in said step (b-1), the conversion into said second signal sample sequence in the integer format is achieved by truncation of a fractional part, and said step (b-4) comprises:

(b-4-1) a step of determining which coding processing is of higher compression efficiency, a first coding processing that codes a range of bits capable of assuming non-zero in a sample of said difference signal in the floating-point format, the range of bits being determined by the bit position of the most significant “1” in the corresponding sample in said second signal sample sequence in the integer format, or a second coding processing that directly codes said difference signal; and

(b-4-2) a step of coding said difference signal by said first coding processing to produce a second code sequence if the result of said determination indicates said first coding processing, or coding said difference signal by said second coding processing to produce a second code sequence if the result of said determination indicates said second coding processing.

Claim 4 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which, in said first coding processing, said range of bits capable of assuming non-zero are losslessly compressed to produce said second code sequence, or said range of bits capable of assuming non-zero are output as said second code sequence without being processed.

Claim 5 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which, in said second coding processing, a mantissa of said difference signal is losslessly compressed to produce said second code sequence, or said mantissa is output as said second code sequence without being processed.

Claim 6 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which the bit length b_1 of one sample in said second signal

sample sequence in the integer format is less than the number of bits b_M of a mantissa of said first signal sample sequence in the floating-point format, and in said first coding processing, said range of bits capable of assuming non-zero is separated into a higher-order sub-range containing a number of bits corresponding to the difference between the number of bits b_M of said mantissa and the number of bits b_I of said one sample and a lower-order sub-range containing the remaining bits, the sub-ranges are separately losslessly compressed to produce respective code sequences, and said code sequences are combined and output as said second code sequence.

Claim 7 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which said first coding processing is performed for each frame containing a plurality of samples and comprises a step of arranging fractional parts, which are ranges of bits capable of assuming non-zero, of mantissas of difference signal samples in each frame with the most significant bits thereof or the least significant bits thereof aligned with each other, selecting a plurality of bits at least in the time-axis direction and/or the amplitude direction to produce a plurality of scanned bit sequences, and losslessly compressing said scanned bit sequences to produce said second code sequence.

Claim 8 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which said first coding processing is performed for each frame containing a plurality of samples and comprises a step of arranging a range of bits capable of assuming non-zero in each of difference signal samples in each frame and scanning the array of said bits in a plurality of different predetermined scanning orders to produce respective scanned bit sequences, a step of coding said scanned bit sequences produced by scanning in said respective scanning orders to produce code sequences for the respective scanning orders,

and a step of determining which scanning order provides the code sequence of the least quantity of information and outputting the determined code sequence as said second code sequence along with an auxiliary code that indicates the corresponding scanning order.

Claim 9 (Original): A lossless coding method for a signal in a floating-point format according to Claim 1 or 3, in which said first coding processing comprises a step of determining, for each frame containing a plurality of samples, the average value of a sequence of integer-value samples, which correspond to the fractional parts of the difference signals in the relevant frame, performing error prediction, in the time-axis direction, about signals resulting from subtraction of the average value from the value of each integer-value sample, losslessly compression-coding the prediction errors, and coding the average value to form said second code sequence.

Claim 10 (Original): A lossless decoding method for a signal in a floating-point format, comprising:

(a) a step of determining whether an input code is a code produced by a single-signal coding or a dual-signal coding, based on an input auxiliary code; and

(b) if the result of said determination indicates the single-signal coding, a step of decoding and expanding an input code sequence as one code sequence to reproduce an original signal sample sequence in the floating-point format;

(c) if the result of said determination indicates the dual-signal coding, a step of separating the input code sequence into a first code sequence and a second code sequence;

(d) a step of decoding and expanding said first code sequence to produce a first signal sample sequence in an integer format;

(e) a step of deriving a difference signal in the floating-point format from said second code sequence;

(f) a step of converting said first signal sample sequence in the integer format into a second signal sample sequence in the floating-point format; and

(g) a step of combining said second signal sample sequence in the floating-point format and said difference signal in the floating-point format to reproduce the original signal sample sequence in the floating-point format.

Claim 11 (Original): A lossless decoding method for a signal in a floating-point format according to Claim 10, in which said step (e) comprises:

a step of determining which of a first expansion method and a second expansion method is used to decode and expand said second code sequence, based on said first signal sample sequence in the integer format or the input auxiliary code; and

a step of decoding and expanding said second code sequence into bit sequences each containing a number of bits capable of assuming non-zero, the number of bits being uniquely determined by the bit position of the most significant “1” in each sample of said first signal sample sequence, and assembling said difference signal from the bit sequences resulting from the decoding and expansion if the result of the determination indicates the first expansion method, or a step of directly decoding and expanding said second code sequence into said difference signal by the second expansion method if the result of the determination indicates the second expansion method.

Claim 12 (Currently Amended): A lossless decoding method for a signal in a floating-point format, comprising:

(a) a step of decoding and expanding a first code sequence to produce a first signal sample sequence in an integer format;

(b) a step of determining which of a first expansion method and a second expansion method is used to decode and expand a second code sequence, based on said first signal sample sequence or an auxiliary code;

(c) if the result of the determination indicates the first expansion method, a step of decoding and expanding said second code sequence into bit sequences each containing a number of bits capable of assuming non-zero, the number of bits being uniquely determined by the bit position of the most significant “1” in each sample of said first signal sample sequence, and assembling the bit sequences resulting from the decoding and expansion into a difference signal;

(d) if the result of the determination indicates the second expansion method, a step of directly producing a digital difference signal in the floating-point format from said second code sequence;

(e) a step of converting said first signal sample sequence in the integer format into a second signal sample sequence in the floating-point format; and

(f) a step of combining said ~~first~~ second signal sample sequence in the floating-point format and said difference signal in the floating-point format to reproduce an original digital signal sequence in the floating-point format.

Claim 13 (Original): A lossless decoding method for a signal in a floating-point format according to Claim 11 or 12, in which the step of producing said difference signal by said first expansion method comprises:

a step of decoding and expanding a first part of said second code sequence into a bit sequence containing a number of bits capable of assuming non-zero, the number of bits being

uniquely determined by the bit position of the most significant “1” in each sample in said first signal sample sequence in the integer format;

a step of decoding and expanding a second part of said second code sequence into a bit sequence containing a number of bits capable of assuming non-zero, the number of bits being uniquely determined by the difference between the number of bits of a sample in said first signal sample sequence in the integer format and the number of bits of a mantissa of an original signal sample sequence in the floating-point format; and

a step of combining the bit sequence resulting from the decoding and expansion of the first part of the second code sequence and the bit sequence resulting from the decoding and expansion of the second part of the second code sequence to produce said difference signal.

Claim 14 (Original): A lossless coder for a signal in a floating-point format, comprising:

first coding means of separating an input first signal sample sequence in the floating-point format into an integer value and an error and losslessly compression-coding the integer value and the error;

second coding means of directly losslessly compression-coding said first signal sample sequence in the floating-point format; and

analyzing and selecting means of determining or estimating which of said first coding means and said second coding means provides higher compression efficiency, selecting the coding means determined or estimated to provide higher compression efficiency, and producing an auxiliary code carrying information about the selection result.

Claim 15 (Original): A lossless coder for a signal in a floating-point format, comprising:

a first coding part that converts an input first signal sample sequence in the floating-point format into a second signal sample sequence in an integer format and losslessly compresses the second signal sample sequence;

a difference producing part that produces a difference signal in the floating-point format that corresponds to the difference between said second signal sample sequence in the integer format and said first signal sample in the floating-point format;

a second coding part that directly codes said difference signal in the floating-point format;

a third coding part that codes a range of bits capable of assuming non-zero in each sample of said difference signal, the range of bits being determined by the bit position of the most significant “1” in the corresponding sample in said signal sample sequence in the integer format; and

means of determining or estimating which of said second coding part and said third coding part provides higher compression efficiency, and selecting and outputting the code from the coding part determined or estimated to provide higher compression efficiency.

Claim 16 (Original): A decoder for a signal in a floating-point format, comprising:

first decoding means of directly losslessly decoding and expanding an input code sequence into a signal sample sequence in the floating-point format;

second decoding means of separating the input code sequence into an integer value and an error and losslessly decoding and expanding the integer value and the error to produce a digital signal sequence in the floating-point format; and

selecting means of selecting one of said first decoding means and said second decoding means based on an input auxiliary code.

Claim 17 (Original): A decoder for a signal in a floating-point format, comprising:

- a first expanding part that decodes and expands an input first code sequence to produce a first signal sample sequence in an integer format;
- a second expanding part that decodes and expands an input second code sequence in a difference signal into a bit sequence containing a number of bits capable of assuming non-zero, the number of bits being determined by the bit position of the most significant “1” in the corresponding sample in said first signal sample sequence in the integer format, and assembles the bits sequences resulting from the decoding and expansion into a difference signal in the floating-point format;
- a third expanding part that directly decodes and expands said input second code sequence into said difference signal in the floating-point format;
- a selecting part that selects one of said second expanding part and said third expanding part based on said first signal sample sequence or an input auxiliary code; and
- a combining part that combines said first signal sample sequence in the floating-point format and said difference signal in the floating-point format received from the selected one of said second expanding part and the third expanding part to produce a third signal sample sequence in the floating-point format.

Claim 18 (Currently Amended): A coding program that makes a computer execute each step of a lossless coding method for a digital signal in a floating-point format according to ~~any one of Claims 1 to 9~~ Claim 1 or 2.

Claim 19 (Currently Amended): A decoding program that makes a computer execute each step of a decoding method for a digital signal in a floating-point format according to ~~any one of Claims 10 to 13~~ Claim 10 or 12.

Claim 20 (Currently Amended): A computer-readable recording medium that stores a program according to Claim ~~18 or 19~~ 1 or 2.

Claim 21 (New): A computer-readable recording medium that stores a program according to Claim 10 or 12.